COMPANY CONFIDENTIAL

GENERAL ELECTRIC

Information Systems Equipment Division
Phoenix, Arizona

NUMBER: 43A219601

Cont. on Sh. 2 Sh. No. 1

TITLE: ENGINEERING PRODUCT SPECIFICATION, PART 1
GE-655 PROCESSOR

REVISION RECORD

REVISION LETTER	DATE	PAGES AFFECTED	APPROVALS	AR OR AN NUMBER
A ISSUED	APR 25 1969			
B	JUN 22 1970	REV. SH. 1 REV. + RETRACED 2-84 RENUMBERED SH 85 TO SH. 92; ADDED SH. 85-91 PMD		ZEB-00-0646
C	MAR 09 1971	REVISED SH. 1, 4, 54, 66 REVISED + RETRACED SH. 55, 65, 81, ADDED SH. 66.1 PMD		ZEB-00-1087
D Rev.	JUN 29 1971	Rev.Sh. 1,6; Ret.& Rev.Sh. 17,18, 19 & 54; Added Sh. 6.1	June 21,1971 PMD	ZEB-00-1196
E Rev.	SEP 27 1971	Rev.Sh. 1 & 19; Ret.& Rev.Sh.18	Sep. 21, 1971 PMD	ZEB-00-1273
F Rev.	NOV 02 1971	Rev. Sh. 1 Ret. & Rev. Shs. 49 & 51 PMD		ZEB-00-1308
G Rev.	JAN 14 1972	Rev.Sh. 1 & 55.	Jan. 10,1972 PMD 72 Jan 12	ZEB-00-1416
H Rev.	JAN 27 1972	Rev.Sh. 1,20,22,23, 55, 66; Ret.& Rev. Sh. 2,4,11,12,24,25, 36,46,47,49,53,68,71, 72,73,75,76; Added Sh. 72.1 thru 72.4 and 75.1 thru 75.3	Jan. 25,1972 PMD 72 Jan 25	ZEB-00-1417

TABLE OF CONTENTS

		Page
1.0	GENERAL DESCRIPTION	7
1.1	Scope	7
1.2	Applicable Documents	7
2.0	FUNCTIONAL CAPABILITIES	8
2.1	Processor Organization	8
2.1.1	Control Unit	8
2.1.2	Operations Unit	9
2.2	Address Preparation and Modification	10
2.2.1	Address Modification	10
2.2.1.1	SCR - Sequence Character Reverse	10
2.2.1.2	Unassigned Tally Designator Codes	10
2.2.2	Base Addressing	11
2.3	Configuration Switching and Interlacing	13
2.3.1	Reasons for Interlacing Storage Addresses	14
2.3.2	655 Active Modules	14
2.3.3	655/635 Active Modules Mixed	16
2.4	655 Processor Performance	17
2.4.1	Configuration and Conditions	17
2.4.2	Instruction Execution Times and Scientific Mix	18
2.4.3	Address Modification Times	20
2.5	Options	21
2.5.1	System Ports and Interlace	21

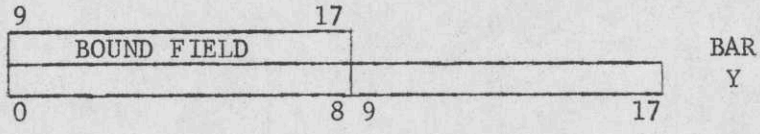
		Page
3.4	635/655 Programming Differences	61
3.4.1	Instructions	62
3.4.2	Miscellaneous	66
3.4.3	Operational Differences Between 635 and 655 Processor	66.1
3.5	Indicators	67
3.6	Configuration Control	67
3.6.1	Address Assignment Switches	67
3.6.2	Interlace Select	67
3.6.3	Size	67
3.6.4	Port Enabling	68
3.6.5	Initialize Control Switches	68
3.6.6	Fault Base Switches	68
3.6.7	Processor Number	68
3.6.8	Initialize and Clear Push Button	68
3.7	Operators Panel	69
3.7.1	Ready/Trouble Indicator	69
3.7.2	Audible Alarm	69
3.8	Maintainability Aids	70
3.8.1	Maintenance Panel	70
3.8.2	History Register	71
3.8.3	Program Accessible T&D Aids	73
3.8.3.1	Fault Register	73
3.8.3.2	Mode Register	75
3.8.4	Additional Aids	75
3.9	Measurement Aids	76
4.0	HARDWARE INTERFACES	77
4.1	System Port Modularity and Interfaces	77

2.2.2 Base Addressing

The 655 shall operate in a manner similar to the 635B or the M-605 in the area of base addressing.

In the slave mode, the processor shall operate with BAR (Base Address Register). The transformation of the effective slave address (Y) to actual store address shall be determined by using the BAR as follows:

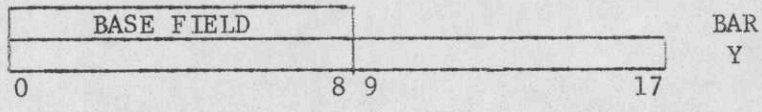
1) A store fault shall occur when $Y_{0-8} \geq BAR_{9-17}$ (bits Y_0 and BAR_9 being the most significant bits).



2) If a store fault does not occur, the storage location used shall be:

$$\sum_{j=0}^{8} \left[(Y_j + BAR_j + C_{j+1}) \times 2^{17-j} \right] + \sum_{j=9}^{17} \left[Y_j \times 2^{17-j} \right]$$

where C_{j+1} is carry from $Y_{j+1} + BAR_{j+1}$ and $C_9 = 0$.



DELETED INFORMATION

This page left blank intentionally.

2.4.3 Address Modification Times

Index register, direct upper, and direct lower address modifications shall have no effect on instruction execution times. For indirect address modification, the instruction execution times shall be modified (for each level of indirectness) up to the maximum shown in Table 2.3.

TABLE 2.3

ADDRESS MODIFICATION TIMES

H

MODIFICATION TYPE	STORE CYCLE TIME	
	0.5 USEC	1.2 USEC
RI	1.1	1.55
IR	1.1	1.55
IT with restoring of the indirect word	2.1	3.0
IT with no restoring of the indirect word	1.1	1.55

3.0 CUSTOMER INTERFACE

3.1 PROGRAM COMPATIBILITY

The 655 Processor shall have all the instructions, modifiers and fault provisions of the GE-615/635. Interrupts shall be recognized after each even-odd instruction pair as in the 615/635. The operation of the slave mode instructions, address modifiers, etc., on the 655 shall be compatible (except for execution time) with the same operations performed on the GE-615/635, provided those instructions and modifiers (and combinations thereof) are specifically listed as allowable in the appropriate reference manuals. New instructions and modifications shall be added to the 655 repertoire. Other instructions shall be changed slightly (see Sections 3.1.1, 3.1.2 and 3.4). B

Any operation code previously used in the GE-605 and 645 shall not be used in the 655 as a new operation. 655 operations that are common to 645 operations shall use the 645 operation code. Any obsoleted GE-615/635 operation code shall not be used for new operations.

Figure 3-1 shows a complete chart of 615/635 operation codes with the additional 655 codes in parenthesis.

3.1.1 Slave Mode

All slave mode instructions, modifiers, etc., of the GE-615/635 shall be provided plus the following:

Address Modifiers

- Sequence character reverse (SCR)

Instructions (see Page 24 for Master Mode)

- Set Zero and Negative Indicator from Storage and Clear (SZNC)*
- Load A and Clear (LDAC)*
- Load Q and Clear (LDQC)*
- Floating Round (FRD)
- Double Precision Floating Round (DFRD)
- Double Precision Floating Store Rounded (DFSTR)

H

* These instructions are required for program gating (see Section 3.1.3).

–	000	001	002	003	004	005	006	007	010	011	012	013	014	015	016	017
000		MME	DRL							NOP				CIOC		
020	ADLX0	ADLX1	ADLX2	ADLX3	ADLX4	ADLX5	ADLX6	ADLX7			(LDQC)	ADL	(LDAC)	ADLA	ADLQ	ADLAQ
040	ASX0	ASX1	ASX2	ASX3	ASX4	ASX5	ASX6	ASX7					ADS	ASA	ASQ	(SSCR)
060	ADX0	ADX1	ADX2	ADX3	ADX4	ADX5	ADX6	ADX7		AWCA	AWCQ	LREG		ADA	ADQ	ADAQ
100	CMPX0	CMPX1	CMPX2	CMPX3	CMPX4	CMPX5	CMPX6	CMPX7		CWL				CMPA	CMPQ	CMPAQ
120	SBLX0	SBLX1	SBLX2	SBLX3	SBLX4	SBLX5	SBLX6	SBLX7						SBLA	SBLQ	SBLAQ
140	SSX0	SSX1	SSX2	SSX3	SSX4	SSX5	SSX6	SSX7						SSA	SSQ	
160	SBX0	SBX1	SBX2	SBX3	SBX4	SBX5	SBX6	SBX7		SWCA	SWCQ			SBA	SBQ	SBAQ
200	CNAX0	CNAX1	CNAX2	CNAX3	CNAX4	CNAX5	CNAX6	CNAX7		CMK			(SZNC)	CNAA	CNAQ	CNAAQ
220	LDX0	LDX1	LDX2	LDX3	LDX4	LDX5	LDX6	LDX7	LBAR	(RSW)		RMCM	SZN	LDA	LDQ	LDAQ
240	ØRSX0	ØRSX1	ØRSX2	ØRSX3	ØRSX4	ØRSX5	ØRSX6	ØRSX7						ØRSA	ØRSQ	
260	ØRX0	ØRX1	ØRX2	ØRX3	ØRX4	ØRX5	ØRX6	ØRX7						ØRA	ØRQ	ØRAQ
300	CANX0	CANX1	CANX2	CANX3	CANX4	CANX5	CANX6	CANX7						CANA	CANQ	CANAQ
320	LCX0	LCX1	LCX2	LCX3	LCX4	LCX5	LCX6	LCX7						LCA	LCQ	LCAQ
340	ANSX0	ANSX1	ANSX2	ANSX3	ANSX4	ANSX5	ANSX6	ANSX7						ANSA	ANSQ	
360	ANX0	ANX1	ANX2	ANX3	ANX4	ANX5	ANX6	ANX7						ANA	ANQ	ANAQ
400		MPF	MPY			CMG				LDE		(RSCR)		ADE		
420		UFM		DUFM		FCMG		DFCMG	FSZN	FLD		DFLD		UFA		DUFA
440	SXL0	SXL1	SXL2	SXL3	SXL4	SXL5	SXL6	SXL7	STZ	SMIC	(SCPR)		STT	FST	STE	DFST
460		FMP		DFMP					FSTR	(FRD)	(DFSTR)	(DFRD)		FAD		DFAD
500	RPL					BCD	DIV	DVF				FNEG		FCMP		DFCMP
520	RPT					FDI		DFDI		NEG		NEGL		UFS		DUFS
540									SBAR	STBA	STBQ	SMCM	STC1			
560	RPD					FDV		DFDV				FMØ		FSB		DFSB
600	TZE	TNZ	TNC	TRC	TMI	TPL		TTF					TEØ	TEU	DIS	TØV
620	EAX0	EAX1	EAX2	EAX3	EAX4	EAX5	EAX6	EAX7	RET				LDI	EAA	EAQ	LDT
640	ERSX0	ERSX1	ERSX2	ERSX3	ERSX4	ERSX5	ERSX6	ERSX7						ERSA	ERSQ	
660	ERX0	ERX1	ERX2	ERX3	ERX4	ERX5	ERX6	ERX7					(LCPR)	ERA	ERQ	ERAQ
700	TSX0	TSX1	TSX2	TSX3	TSX4	TSX5	TSX6	TSX7	TRA					TSS	XEC	XED
720	LXL0	LXL1	LXL2	LXL3	LXL4	LXL5	LXL6	LXL7		ARS	QRS	LRS		ALS	QLS	LLS
740	STX0	STX1	STX2	STX3	STX4	STX5	STX6	STX7	STC2	STCA	STCQ	SREG	STI	STA	STQ	STAQ
760										ARL	QRL	LRL	CTB	ALR	QLR	LLR
	000	001	002	003	004	005	006	007	010	011	012	013	014	015	016	017

H

H

GE-655 INSTRUCTION MNEMONICS
CORRELATED WITH THEIR OPERATION CODES

Figure 3-1

3.1.1 Slave Mode (Continued)

All unassigned instruction codes and all address modifiers expressed as exceptions shall be considered as illegal codes. Execution of these illegal codes shall result in an Illegal Procedure Fault (Section 3.3.3).

Certain illegal sequences of instructions shall result in an Illegal Procedure Fault. Most of these sequences are combinations or REPEAT type instructions with other instructions (Section 3.3.3.4).

The Illegal Procedure Fault shall occur whenever anything not defined as legal is attempted. It shall therefore enforce restrictions on the use of instructions that, in some instances, in the 615/635 were only forbidden by the documentation.

The Timer Register in the 655 shall have increased resolution requiring changes to the instructions STT and LDT (master mode). See the 635/655 instruction differences (Section 3.4.1).

The floating store rounded, single precision, instruction (FSTR) shall be changed from the 615/635 in that a true round shall be performed and the zero and negative indicators shall be set. See the 635/655 instruction differences (Section 3.4.1). A true round is defined in Section 3.2.1, pp. 31, 32, 33 & 34.

3.1.2 Master Mode

All master mode instructions, modifiers, etc., of the 615/635 shall be provided with the following additions and modifications:

Additions to the instructions included from the 615/635 Master Mode list shall be as follows:

- Read System Controller Registers (RSCR)
- Set System Controller Registers (SSCR)
- Store Central Processor Registers (SCPR)
- Load Central Processor Registers (LCPR)

3.1.2 Master Mode (Continued)

Changes/additions to the 615/635 repertoire shall be:

- CIOC, Connect I/O Port. With the 655 System Controller, the C(Y) shall not be transmitted to the I/O system port on this execution (C(Y) not used by I/O system port in 635). Only the connect ($CON) strobe shall be sent through the system port selected by bits 33-35 of the C(Y). Data transactions between an active module and the 655 System Controller shall be accomplished via a bidirectional data bus. Therefore, the bus may be busy per a system module request which prohibits use of the bus for connect word transmission. With the 635 System Controller, the CIOC instruction is identical with the 615/635.

DELETED INFORMATION

This page left blank intentionally.

DELETED INFORMATION

This page left blank intentionally.

DELETED INFORMATION

This page left blank intentionally.

3.2.3 Special and Diagnostic Aid Instructions

In the 655, there shall be a Read Switches (RSW) instruction to be used as a special aid (see 3.2.2) with certain master mode instructions (ex. RSCR) and as a diagnostic aid to allow maintenance personnel to select, on the fly, options in their test programs. To allow maximum usage of this instruction as a diagnostic aid, it shall be legal in the slave mode. In addition, two monitor instructions (PULS1 and PULS2) may be "kludged" into the 655 (see 3.9). It should be noted, however, that any instruction that requires manual intervention (i.e., throw switches) for effective usage or that generate signals to monitors, should under no circumstances be a common usage instruction in a multiprogramming environment. For this reason, this instruction should only be documented in limited access areas (ex. this EPS-1) and should be left out of general user documentations (ex. Programmer's Reference Manual).

The RSE instruction shall work as follows:

Mnemonic:	Name of the Instruction:	Op Code (octal)
RSW	Read Switches	231

Summary: This instruction shall cause the processor data/configuration/fault base, processor number switches, and BAR mode to be read into the A register. The particular switches to be read shall be determined by the address field of the instruction. The address assignments (only use last 2 bits) shall be as follows:

Address (octal)	Meaning
XXXXX0	36 data switches on the processor panel
XXXXX1	Configuration switches
XXXXX2	Fault base, BAR mode and processor number switches

Modifications: All allowed

Indicators: None affected

Note: The format of the configuration switches shall be as follows:

For RSW address (octal) of XXXXX0 (or XXXXX3):

36 Data Switches on the Processor Panel
0 35

3.3.3.1 Illegal Operation Code (Continued)

- The fault shall occur on any instruction containing bit 27 = 1 or bit 29 = 1.

3.3.3.2 Illegal Modifiers

The illegal use of modifier shall cause the Illegal Procedure fault as follows: (except for SCR, the symbols used are described in the 625/635 Programming Reference Manual, pp 34-36).

- The fault shall occur if an unassigned IT tag designator is detected (t_d = 0001, 0010, 0011, 0110, 0111).
- The fault shall occur if DU or DL is specified as the original register designator with the RI modification type.
- The fault shall occur if the indirect word fetched using IDS or DIC as the tally designator specifies an R or RI modifier, where the register designator is not N.
- The fault shall occur if a repeated instruction has a modifier other than R or RI, or if R or RI is specified and the register designator is not X1 - X7.
- The fault shall occur if a Repeat Link is attempted with a modifier of other than R.
- The fault shall occur if the instructions SCPR or LCPR are attempted with a tag of other than those assigned.

3.3.4 Program Generated Faults

These faults shall be identical with the 615/635 except for the LUF (Lock-up) Fault, the FCMD (command) fault and the STR (Store, formerly FMEM memory) fault. The changes to these faults are as follows.

3.3.4.1 LUF (Lock-UP) Fault

[G] The lock-up time shall be specified by the Lock-Up Fault Register*which can be program loaded using the LCPR instruction. The register shall be specified in the tag field. Lock Up times of 2, 4, 8 or 16 milliseconds may be specified as follows:

Lock-Up Time	Code in LUF Register
2	00
4	01
8	10
16	11

This time will not be subject to the same kind of uncertainty as the 615/635. That is, when the time is set to 4 milliseconds, the LUF time will be exactly 4 milliseconds and not somewhere between 4 and 8 milliseconds. Otherwise, the LUF shall function identically with the 615/635.

3.3.4.2 FCMD (Command) Fault

The following changes from the 615/635 shall be made:

[H] o The following additional instructions shall generate a FCMD fault when attempted in slave mode: DIS, LBAR, LCPR, LDT, RSCR, SCPR, SSCR.

o An attempt by the processor to execute the SMCM, SSCR, or the XEC instructions without an Interrupt Mask Register assigned to the System Controller Port to which the processor is connected shall generate an FCMD fault.

o A conditional illegal action may be set to cause the FCMD fault using the switches on the System Controller maintenance panel.

[G] *The lock-up time specified by the Lock-Up Fault Register is for Slave mode only. During Master mode a value of 32 milliseconds will be used. If during entirely inhibited Master mode servicing, the Slave mode LUF time is exceeded (and the Master mode LUF is not), the Slave program will be aborted upon completion of the Master mode servicing.

3.4.2 Miscellaneous

ITEM	635 MANUAL	635 ATTEMPTED EXECUTION	655
Timer	24 bits lsb (23) - 1/64 millisecond		27 bits lsb (26) - 1/512 millisecond
Instruction Bits 27 29	Must be zero	Have no effect on operation	A one in either bit causes an IPR fault
Unassigned IT tags	Not mentioned	Detection causes a ZOP fault	Detection causes an IPR fault
DU* DL*	Illegal	Indirect function ignored	These tags cause an IPR fault
R Mod RI Mod IR Mod IT-I Mod -CI -F	No Change		
IT-ID -DI -IDC -DIC -AD -SD -SC	Executed with Read-Alter-Rewrite Cycle		Executed with Read Cycle followed by Write Cycle
IT-SCR	Not implemented		New IT Modifier
Execute Interrupts	Not Mentioned	Port Address Switches (see 2.3.2) used in bits 9-11 of effective address for XED	Bits 9-11 of effective address for XEP assumed 000.

3.6.4 Port Enabling

A port shall be enabled or disabled with a PORT ENABLE switch, on the configuration panel: one switch, three positions for each port. The port shall be enabled, disabled, or under program control.

3.6.5 Initialize Control Switches

The switches shall indicate whether the processor will accept the system initialize signal from the port. If ON (=1), the system initialize signal shall be accepted when the system controller generates it. If OFF (=0), any initialize from that port shall be ignored. Disabling the port shall override the Initialize Control Switch.

3.6.6 Fault Base Switches

These seven switches shall supply address bits 6 - 12 of all fault vectors as described in the 625/635 Programming Reference Manual, page 20. These switches shall be program accessible by the RSW instruction.

3.6.7 Processor Number

The two processor number switches shall indicate the number assigned to the processor (in a multiprocessor system). They shall have no affect on the processor logic, but shall provide a program accessible number unique to the processor, readable by the RSW instruction.

3.6.8 Initialize and Clear Push Button

The Initialize and Clear Push Button shall cause all control bistable and data registers in the processor to be reset to a known initialized state. This button shall cause the same action as the Initialize signal through a system port.

3.8.1 Maintenance Panel (continued)

- Stop-on-Fault selection shall be provided.
- An Inhibit Fault control for the following faults shall be provided:
  - -- Start-Up Fault (SUF)
  - -- Operation Not Completed Fault (FONC)
  - -- Lock-Up Fault (LUF)
  - -- Parity Fault (FPAR)
  - -- Command Fault (FCMD)
  - -- Store Fault (STF)
  - -- Connect Fault (CON)
  - -- Timer Runout Fault (TROF)
  - -- Shut-Down Fault (SDF)
- An Initialize push button shall cause all control bistables to be reset to a known state. Data registers shall not be affected.
- A manual lamp test switch shall be provided.

3.8.2 History Register

The purpose of the History Registers is to reflect the state of the processor during the last 16 steps which it performed. When an unexpected fault or like condition occurs, the History Registers shall contain data which reflects what the processor was doing as far as 16 operations back. These registers shall be interrogatable by a diagnostic program. In addition, any of the 32 registers shall be selectable from the Maintenance Panel and be displayed.

The History Register shall contain 32 (72 bit) registers. 16 of the registers shall be controlled by the Control Unit (CU) and contain historical data pertaining to the CU. The other 16 registers shall be controlled by the Operations Unit (OU) and shall contain historical data pertaining to the OU. Operation of the History Registers shall be controlled by the Mode Register (see 3.8.3). Data shall be written into the CU registers at the end of every operand or instruction fetch address preparation cycle. Data shall be written into the OU registers at the end of the last cycle of an OU operation. Each of the two groups of 16 registers shall have associated with it, a 4-bit counter which selects one register within the group of 16. The contents of these counters shall be displayable on the maintenance panel. Prior to the information being strobed into the selected register, the counter shall point to the oldest data in the history register.

3.8.2 History Register (continued)

In order to read the History Register and place it in the store the SCPR instruction must be used. The tag field of the SCPR shall indicate which of the 16 register groups will be stored, the CU group or the OU group. Each time an SCPR instruction is executed, the counter associated with the group of 16 registers specified by the tag, shall be incremented by one. Therefore, in order to store a full group of 16 registers, an SCPR instruction must be executed by the Processor 16 times in succession.

History Register Data Format

The data fields in each of the 16 CU and 16 OU History Registers are shown below:

CU Registers

0	18	36	53	58	62	69	71
CU Control Flags	Op Code and Tag	Address REG 0-17		CMD REG	Port Select	Function Buffer	Port Logic

where:

00-17 CU Control Flags

Tells what the CU is presently doing.

Bit	Flag	A "1" Means:
0	PIA	Preparing Instruction Address
1	POA	Preparing Operand Address
2	RIW	Requesting Indirect Word
3	SIW	Restoring Indirect Word
4	POT	Preparing Operand Tally
5	PON	Preparing Operand Address - Ignore further indirection
6	RAW	Requesting Alter Rewrite Word
7	SAW	Restoring Alter Rewrite Word
8	TRGO	Transfer Go (conditions met)
9	XDE	Doing an XED from an Even Location
10	XDO	Doing an XED from an Odd Location
11	IC	Executing the Odd location of an instruction pair
12	RPTS	Doing a Repeat Operation

3.8.2 History Register (continued)

Bit	Flag	A "1" Means:
13	WI	Waiting for (fetching) an Instruction
14	AR F/E	Address Register contains valid information
15	$\overline{\text{XIP*ADR/ZFA}}$	NOT preparing an XIP Address
16	$\overline{\text{FLT*ADR/ZFA}}$	NOT preparing an FLT Address
17	$\overline{\text{ADD BASE}}$	NOT in slave mode

18-35 Op Code and Tag

A copy of bits 18-35 of the Instruction being executed.

36-53 Address Register

A copy of bits 00-17 of the Address Register.

54-58 Command Register

A copy of the Processor Command Register

59-62 Port Select

A copy of the Port Select Lines

63-69 Function Buffer

Tells what the CU has done and the port operation that is starting.

Bit	Function	A "1" Means:
63	XEC-INT	An execute interrupt
64	INS-FETCH	An instruction fetch
65	CU-STORE	Load or store on behalf of the CU or OU
66	OU-STORE	
67	CU-LOAD	
68	OU-LOAD	
69	DIRECT	A direct cycle

70-71 Port Logic

Tells the status of the port logic

70	$\overline{\text{PC BUSY}}$	Port control logic not busy
71	Port BUSY	Port interface busy.

3.8.2 History Register (continued)

OU Registers:

where:

00-16 Primary OU Register (RP)

Holds next instruction Op code and sets up input data switches.

Bits	Information:
0-8	Bits 18-26 of the next instruction to be executed (Op Code)
9	Bit 30 of IT tally word (6/9 bit data)
10-12	Bits 33-35 of the next instruction to be executed (either latter part of the tag or the character #)
13	Indicates character modification (IT)
14	Indicates direct operation (DL = 10°14, DU = $\overline{10}$·14)
15,16	LREG/SREG Effective Address Counter
17	Spare
18-26	Secondary OU Register (RS)

A copy of bits 0-8 of the RP register. It is strobed when the input data is strobed into the OU. It is used to do the actual instruction execution.

27-39 OU Control Flags

It tells the state of the RB1, RP, & RS registers and tells what OU cycle has been executed.

Bit	Function	A "1" Means:
27	FRB1-FULL	OU OPCODE buffer Full
28	FRP-FULL	Primary Register Full
29	FRS-FULL	Secondary Register Full
30	FGIN	First Cycle for all OU OPs
31	FGOS	Second Cycle for OU Multiple OPs
32	FGD1	First Divide Cycle
33	FGD2	Second Divide Cycle
34	FGOE	Exponent Compare Cycle

3.8.2 History Register (continued)

Bit	Function	A "1" Means:
35	FG0A	Mantissa Alignment Cycle
36	FG0M	General OU Cycle
37	FG0N	Normalize Cycle
38	FG0F	Final Cycle
39	FSTR-OP-AV	OU store data available (reset by CU)

40-50 Registers

Tell which registers are NOT being used.

Bit	Register
40	DATA NOT available
41	A reg NOT in use
42	Q reg NOT in use
43	X0 reg NOT in use
44	X1 reg NOT in use
45	X2 reg NOT in use
46	X3 reg NOT in use
47	X4 reg NOT in use
48	X5 reg NOT in use
49	X6 reg NOT in use
50	X7 reg NOT in use

51-53 Spare

54-71 ICT Tracker

Tells the address of the OU instruction (skips CU instructions and therefore can point further back than 16 instructions).

3.8.2 History Register (continued)

The tag field of the SCPR instruction shall specify which group of 16 registers shall be stored, hence address modification cannot be allowed on this instruction. The tag field code is shown below.

Select CU History Register - tag field = 010000 (20_8)

Select OU History Register - tag field = 100000 (40_8)

Since the function of the History Register is to provide historical data pertaining to the state of the processor at the time of an unexpected fault condition and prior to it, there must be a means of inhibiting writing into it until after the register has been interrogated. The conditions for applying this inhibit are as follows:

1) Inhibit the write strobe for the CU-HR and the OU-HR unless bit 30 (and 35) in the mode register is set. (This is the only inhibit on the OU-HR write strobe.)

2) Inhibit the CU-HR write strobe if either the address trap, op-code trap or CU-HR counter = 15 trap is enabled and the condition is met. These traps are a function of bits contained in the mode register.

3) Inhibit the CU-HR write strobe if either the CU-HR or OU-HR is being stored via an SCPR instruction. This is to prevent the register from changing while it is being stored. (There is no need to inhibit the OU-HR write strobe because an SCPR is not an OU type instruction.)

4) Inhibit the CU-HR write strobe, if bit 29 of the mode register is set and there is no match between the op-code specified by bits 18-26 of the mode register.

5) Inhibit the CU-HR write strobe when doing an LCPR with a tag of 3 or 7.

6) In an Op Not Complete fault or, if bit 31 of the mode register is set, Lock Up, Parity, Command Store, Illegal Procedure, Shutdown, Op Code Trap, CU-HR counter overflow trap, or address match trap occurs.

3.8.2 History Register (Continued)

[H]

When the HR write inhibit condition exists, the select counters shall not be advanced, except on an SCPR instruction.

In order to verify that the HR is functioning correctly the ability to load known data into the HR and then read it, for comparison, shall be provided. The LCPR instruction shall be provided with two tag codes for this purpose, $(03)_8$ for loading all zeros and $(07)_8$ for loading all ones.

In order to completely fill the CU and OU History Registers with known data it shall be necessary to execute the LCPR instruction, with one of the tag codes mentioned, 16 times in succession. The register can then be stored by the execution of 16 SCPR instructions. Verification of data can then be made.

3.8.3 Program Accessible T&D Aids

[B]

The following program accessible registers shall be provided as T&D aids in addition to the History Registers:

- Fault Register
- Mode Register

3.8.3.1 Fault Register

The fault code register shall be used to differentiate ambiguous error indications such as FPAR, IPR, etc. Every error detected by the processor which results in a fault trap shall have a separate and unique bit or code set in the fault code register, such that more than one fault indication may be in the register at the same time. The action of setting a new fault code (for a different fault) shall not destroy the previous fault indications.

The bit format is shown in Table 3-3.

The fault register shall be used by the SCPR with a tag of octal 1. After the contents of the register are stored, it shall be cleared to zeros.

3.8.3.2 Mode Register

The mode register shall provide the program control of the T&D hardware aids. It shall be loaded by executing an LCPR instruction with a tag of $(04)_8$ and stored by executing an SCPR instruction with a tag of $(06)_8$. (See 3.2.2.)

The following is the definition of the 36 bits of the Mode Register:

0-14 Floating fault vector.

These bits shall constitute the upper 15 bits for the fault pair addresses of the various fault traps enabled by the Mode Register (see bits 17, 27, and 28). Bits 15 and 16 of these addresses are hard wired according to the trap condition encountered and are specified below. Bit 17 is assumed zero.

MR bit	trap condition	Vector Address 15	16
--	not used	0	0
27	Op Code trap	0	1
28	CU-HR Counter Overflow	1	0
17	Address match trap	1	1

15, 16 Unassigned

17 Trap on Address match.

If bit 17 = 1 and the address in the Processor Address Register matches the address switches on the maintenance panel, the Processor shall do an XED to the fourth of the fault pairs specified by the Floating fault vector (bits 15, 16 = 1, 1).

18-26 Op Code/Software Switches

If either bit 27 or bit 29 = 1, bits 18-26 shall represent an Op code on which the Processor shall trap (see bit 27) or on which it shall strobe the CU History register (see bit 29).

3.8.3.2 Mode Register (continued)

If both bit 27 and bit 29 = 0, bits 18-26 shall provide the means of software "setting" certain of the Maintenance Panel switches:

$18 \cdot \overline{27} \cdot \overline{29}$ Set CU overlap inhibit. The CU shall wait for the OU to complete its operation on the even instruction operand before it starts the corresponding odd instruction operand. The CU shall also wait for the OU to complete its operation on the odd instruction operand before it fetches the next instruction pair.

$19 \cdot \overline{27} \cdot \overline{29}$ Set Store overlap inhibit. The CU shall wait for the $DA before it issues the next interrupt.

$20 \cdot \overline{27} \cdot \overline{29}$ Set Store Incorrect Data Parity. The CU shall cause incorrect data parity to be sent to the Store for the next store instruction and then reset bit 20.

$21 \cdot \overline{27} \cdot \overline{29}$ Set Store Incorrect ZAC Parity. The CU shall cause incorrect ZAC parity to be generated on each memory cycle until the $DA of the next store instruction. At this time bit 20 will be reset.

$22 \cdot \overline{27} \cdot \overline{29}$
$23 \cdot 27 \cdot 29$ Set Timing Margin (if the manual switch is in the Normal position) accordingly:

	22	23
Slow	0	1
Fast	1	1
Normal	X	0

$24 \cdot \overline{27} \cdot \overline{29}$
$25 \cdot \overline{27} \cdot \overline{29}$ } Unassigned
$26 \cdot \overline{27} \cdot \overline{29}$

27 Trap on Op code

If bit 27 = 1 and the Op code of the instruction whose address is presently being prepared by the Processor (including indirect cycles) matches bits 18-26 of the Mode Register, the Processor shall do an XED to the second of the fault pairs specified by the Floating fault vector (bits 15, 16 = 0, 1).

3.8.3.2 Mode Register (continued)

28 Trap on CU History Register counter overflow.

If bit 28 = 1 and the CU-HR counter overflows*, the Processor shall do an XED to the third of the fault pairs specified by the Floating fault vector (bits 15, 16 = 1, 0).

These three traps (Address, Op code, CU-HR counter overflow) shall occur after the next complete odd instruction following their detection. They shall be treated as Group V faults as far as when serviced and when inhibited. Their individual priorities shall be:

1 - CON
2 - TROF
3 - SDF
} Other Group V faults
4 - Op Code trap
5 - CU-HR Counter Overflow
6 - Address match trap
7 - External Interrupts

29 Strobe the CU-HR on Op code.

If bits 29 and 30 (see below) are = 1 and the Op code of the instruction whose address is presently being prepared by the Processor (including indirect cycles) matches bits 18-26 of the Mode Register, the CU-HR will be strobed.

30 Enable History Registers

If bit 30 = 1, the CU and OU History Registers may be strobed. If bit 30 = 0 or bit 35 = 0 (see below), they will be locked out. This bit will be reset by either an LCPR with the bit corresponding to 30 = 0 or by an Op Not Complete fault. It may be reset by other faults (see bit 31). After being reset, it must be enabled by another LCPR instruction before the History Registers may be strobed again.

- - -

*This means that the CU-HR may be locked out for the latter indirect cycles of an instruction. Also, setting bit 28 shall cause the CU-HR counter to be reset to 0.

3.8.3.2 Mode Register (continued)

31 Additional resetting of bit 30.

If bit 31 = 1, the following faults will also reset bit 30:

- Lock Up
- Parity
- Command
- Store
- Illegal Procedure

- Shutdown
- Op code trap (bit 27)
- CU-HR counter overflow trap (bit 28)
- Address match trap (bit 17)

32-34 Unassigned

35 Use Mode Register.

Unless bit 35 = 1, all other bits in the Mode Register will be ignored and the History Registers shall be locked.

3.8.4 Additional Aids

(The following aids are listed and are to be further defined.)

o Various illegal machine states shall be detected and cause the FONC fault. (This is called finger pointing logic.) For example, the store unit interface sequence checks: limited echo checks on minor internal interfaces.

o Automatic external parity error indicator that can only be reset manually.

3.9 MEASUREMENT AIDS

The following aids to system measurement shall be provided:

- High resolution (1/512 millisecond) in the clocks, Timer Register and Elapsed Time Clock. (The ETC shall be read by the processor but is located in the 655 System Controller.)
- The means to connect a freestanding monitor.

The Monitor, when present in the system, shall be used to detect, record and process selected system events. Examples (not all inclusive) of system events in the processor which may be used in system measurement are as follows:

- Instruction type occurrence in a program
- Transfer occurrence
- I/O initiation (CIOC)
- Mode change (master/slave)
- Interrupts
- Faults

Implementation of the Monitor shall require a hardware "kludge" of special processor instructions to provide counting signals.

The following instruction operation codes shall therefore be reserved for future Monitor use and shall not be assigned to another function:

Octal Operation Code	Mnemonic
012	PULS1
013	PULS2